1-1-1983

Report of the joint data base task force

American Institute of Certified Public Accountants;Canadian Institute of Chartered Accountants;Institute of Internal Auditors

REPORT OF THE JOINT DATA BASE TASK FORCE

AMERICAN INSTITUTE OF CERTIFIED PUBLIC ACCOUNTANTS

CANADIAN INSTITUTE OF CHARTERED ACCOUNTANTS

INSTITUTE OF INTERNAL AUDITORS

REPORT OF THE JOINT DATA BASE TASK FORCE

AMERICAN INSTITUTE OF CERTIFIED PUBLIC ACCOUNTANTS

CANADIAN INSTITUTE OF CHARTERED ACCOUNTANTS

INSTITUTE OF INTERNAL AUDITORS

NOTICE TO READERS

The report has been prepared by a task force composed of members of the American Institute of Certified Public Accountants, the Canadian Institute of Chartered Accountants, and the Institute of Internal Auditors. The views expressed are those of the task force members and have not been approved or endorsed by any committee, governing body, or membership of the sponsoring Institutes.

Joint Data Base Task Force

Walter D. Pugh,
Chairman (AICPA)
Timothy R. Bertram (IIA)
P.J. Corum (IIA)
Raymond H. Healey (CICA)
William C. Mair (AICPA)
Robert G. Parker (CICA)
Michael W.R. Stoneham (CICA)

Published by the AICPA on behalf of the three
participating organizations

American Institute of Certified Public Accountants, Inc.
1211 Avenue of the Americas, New York, New York 10036-8775

The Canadian Institute of Chartered Accountants
150 Bloor Street West, Toronto, Ontario M5S 2Y2

The Institute of Internal Auditors, Inc.
249 Maitland Avenue, Altamonte Springs, Florida 32701

1 2 3 4 5 6 7 8 9 AudS 8 9 8 7 6 5 4 3

Introduction

A data base environment can affect the system of internal control and influence the nature, timing, and extent of audit procedures. To study these effects the American Institute of Certified Public Accountants, the Canadian Institute of Chartered Accountants, and the Institute of Internal Auditors formed a task force.

This report, which is aimed at general management and financial auditors, has been compiled from the task force's findings. It describes the effects a data base environment may have on control and audit procedures, but it is not a tutorial or a technical examination of data base technology.

Many people use the terms *data base, data base management system,* and *data base environment* interchangeably. They are not synonymous, however, and a clear distinction is necessary to identify the impact of this technology on internal control and auditing. The task force uses these three terms as follows:

Data base. Any collection of related information.

Data base management system (DBMS). A sophisticated set of software products that facilitates the creation, management, maintenance, security, and control of a data base and provides the interface between application programs and the data base.

Data base environment. An environment in which a data base, a DBMS, and a data processing environment exist and in which there is an appreciable level of data sharing among diverse users.

A basic level of data processing knowledge concepts and terminology is presumed on the part of the reader even though the use of jargon and technical terms is limited and the depth of technical discussion is restricted. Some liberties have been taken in terminology and definitions, but, in most instances, when technical terminology is used the Conference on Data Systems Language (CODASYL) definitions apply.

The report is divided into three chapters.

1. "The Data Base Environment" provides an overview of the features and characteristics of a data base environment.
2. "Control Considerations in a Data Base Environment" identifies those characteristics in a data base environment that are different from a traditional EDP environment and that may affect the system of control.
3. "Audit Considerations in a Data Base Environment" provides guidance to the auditor on the study and evaluation of control and suggested audit procedures.

Contents

1 The Data Base Environment

There is considerable technical literature that describes the data base environment and the operation of data base management systems (DBMSs). The purpose of this chapter is to present an overview of the significant concepts as a framework for considering control and audit implications.

Data can exist in different formats in a computer environment. The simplest are those that are designed for a specific application. In such an environment an application is related to a file or group of files. Generally, the way the data are physically stored is the way the data are used by the application program. If another application program is to share the data, it may be necessary to physically reorganize or resequence the file. Files can be reorganized by sorting, merging, accessing multiple files, creating extract files, or by using program logic.

A more complex organization of data uses a software product (a DBMS) to structure the data. The DBMS allows multiple users to access specified items of data, assembled in the way each program wishes to view the data. The data base environment as defined in this report requires the use of a DBMS, usually a commercial software product; however, the necessary software can also be developed in-house.

Overview of a Data Base Management System

A data base is defined as a collection of related information that is shared and used by a number of different users for different purposes. Each user may not be aware of all the types of available information or the ways in which the data can be assembled for multiple purposes.

A data base is composed of related data elements, which are sometimes organized into record types or segments. A data element is the basic unit of data and cannot be subdivided into smaller data types. For example, an employee name could be one data element;

the same employee's ID number could be a separate data element. A record for this employee could consist of several related data elements. The DBMS permits data elements to be accessed and manipulated for application programs in the unique order and sequence required by the application.

All the data elements in a data base and the physical relationships in which they are stored are referred to as the *physical view* of the data, or the *schema;* a subset of the data base that a user is authorized to access or update is called a *logical view,* or *sub-schema.* The principal attributes of a sub-schema are order, content, and function.

Order. The particular sequence in which the data are presented. For example, a sub-schema containing a customer's account number, name, sales order, and sales representative number and name would form the basis for an order entry system application. The same data accessed by another program sorted by sales representative, sales order, and customer would support a sales commission system.

Content. The specific data elements contained in the data base that the sub-schema requires to be assembled.

Function. The type of actions that the application program is permitted to perform on the assembled data. For example, one sub-schema may present customers, orders, and sales representatives, in that order, and allow its application program only to read the data. A different sub-schema may present the same data in the same order but allow its associated application program to read, update, and delete the data.

The data elements are accessed through the programmed input/output features of the DBMS so that they are independent of the application program that uses them. One user may perceive a sub-schema of data elements in the format required by an application in quite a different way from another user's perception, even though both users are accessing the same data. For example, one user of a payroll data base may perceive the employee data to be organized by cost center and then alphabetically within the cost center. A second user may perceive the data base in strict alphabetical order by employee with cost centers indicated for each employee. However, the data may be physically stored in some third order, such as in ascending social security number sequence.

The data elements that compose the data base can be organized into one of several different structures, which generally fall into two categories: hierarchies and networks. A hierarchical data structure is one in which the data relationships follow a tree-branching structure. That is, each subordinate data element can have only one "parent"

data element, but a parent data element type can have any number of subordinate ("child") elements. A special type of hierarchical data structure is one that contains only one record type and all the data elements within that record type are related to each other on a one-to-one basis. This data structure is referred to as a single sequential "flat file," or relational data structure.

When a data structure diverges from this form to allow a record type to have more than one parent, the type of structure can no longer be categorized as a hierarchy and is referred to as a network structure.

There are several methods for maintaining the hierarchies or networks defined by the schema and sub-schema. All methods involve the use of *pointers*. A pointer is the address of the physical location of data; the pointer allows a DBMS to access the "pointed to" data and locate and assemble the data elements making up the record required by the application program. One method for maintaining a hierarchy or network structure is for each data element to include a pointer to the next associated data element; this is generally referred to as an *imbedded pointer* method. Another method uses an index that is a list of identifiers of data elements containing some common attribute and their associated physical location. These methods are used to store or retrieve a particular data element.

In addition to maintaining the interrelationships between the data elements (and records), extensive tables, often referred to as data directories, are maintained by the DBMS to indicate what data elements are required by each of the users and in which order. The data directories also contain information about the physical location of the data.

Having introduced a number of concepts, an explanation of the basic way in which a DBMS functions with a typical application program may be useful at this point. Following are the principal steps in order of their occurrence.

1. A sub-schema would be set up to define the data required and the appropriate order for retrieval by an application program. This is done during programming without concern for the physical location of the data that will be acted upon by the program.
2. A sub-schema would be requested by an application program. The sub-schema that is recorded in the DBMS directory is used each time the program is run. The sub-schema is compared by the DBMS to the directory of current and valid sub-schemas. If the application program and requested sub-schema match the information in the directory, the program can continue to be processed.
3. When the application program is required to read, update, or delete data, the DBMS is accessed. The location of the data within

the data base is under the control of the DBMS and is essentially independent of the application program. The data specified are located within the data base and provided by the DBMS to the application program for processing (that is, read, update, delete, and so on).

Unique Features of a Data Base Environment

The impact of the data base environment on control is distinguished by two important features. These features relate to *how* data are being accessed (data independence) and *what* is being done (data sharing). The following paragraphs discuss data independence and data sharing.

Data Independence

Data independence is accomplished when a single physical representation of the data is used to satisfy requests for data from multiple application programs. This independence is achieved by separating the application program from the physical data storage while maintaining the logical relationship through the DBMS. The DBMS maintains the definition of the logical views (sub-schemas) required by each application and constructs the sub-schema requested from the physical representation of the data.

DBMSs differ in the degree of data independence they provide. A major goal of DBMS research and development is to provide greater data independence. The degree of data independence achieved helps determine the ease with which the data processing department can accomplish changes to application programs. Complete logical data independence will be achieved only when the physical structure of the data base can be changed without altering application programs, thus separating completely the application programs from the physical data structures.

Data Sharing

Individual application programs require the same data elements for different purposes. For example, one program may require the customer open invoices to perform a credit check, while another program may require all the invoices to perform a market analysis.

The greater the number of different logical views of the same data, the greater the degree of data sharing. Although a high degree of data sharing is possible without the use of a DBMS, the primary impact of

the DBMS is to facilitate sharing on a data element level rather than record level through separate sub-schemas. In addition, the use of a DBMS reduces the degree of technical expertise required on the part of the application programmer.

Characteristics of a Data Base Management System

Ease of Program Maintenance

Segregating data from application programs with the aid of a DBMS facilitates program maintenance. To illustrate this concept, figure 1 shows a traditional data processing maintenance cycle, in which three users supply data to four application programs. These programs access four files that contain twelve data items. Each application program was written to support a particular function.

If user II requires a change in program 3 so that it incorporates a new data item, N, program 3 must be changed. In addition, data item N must be added to a file. If this file is used by other programs they must also be changed. In the example both programs 1 and 2 will also require amendment. Therefore, a functional change to one program has triggered three programming changes and the need for a one-time file-conversion program.

Figure 2 illustrates the data base environment equivalent; there are three users and four application programs. Instead of separate files supporting individual applications all the data have been collected into a single data base designed to support all three users.

In this situation, introducing an additional data item N for user II will only require amendment to program 3 and its related sub-schema. All programs access data via the DBMS and can only access the data they require. Since programs 1, 2, and 4 do not access data item N, they do not require amendment.

Access and Security

Computer systems contain information that is essential to a company's operations. Inadvertent or intentional loss or alteration of data can impair a company's ability to continue normal operations. Likewise, disclosure of data can be competitively or legally damaging. These situations are recognized in most DBMSs, and features are provided to control access and enhance security beyond that provided by the operating system. Such features usually entail restricting use of the data base. It is important to recognize that these features are optional

Figure 1
Traditional
Data Processing
Maintenance Cycle

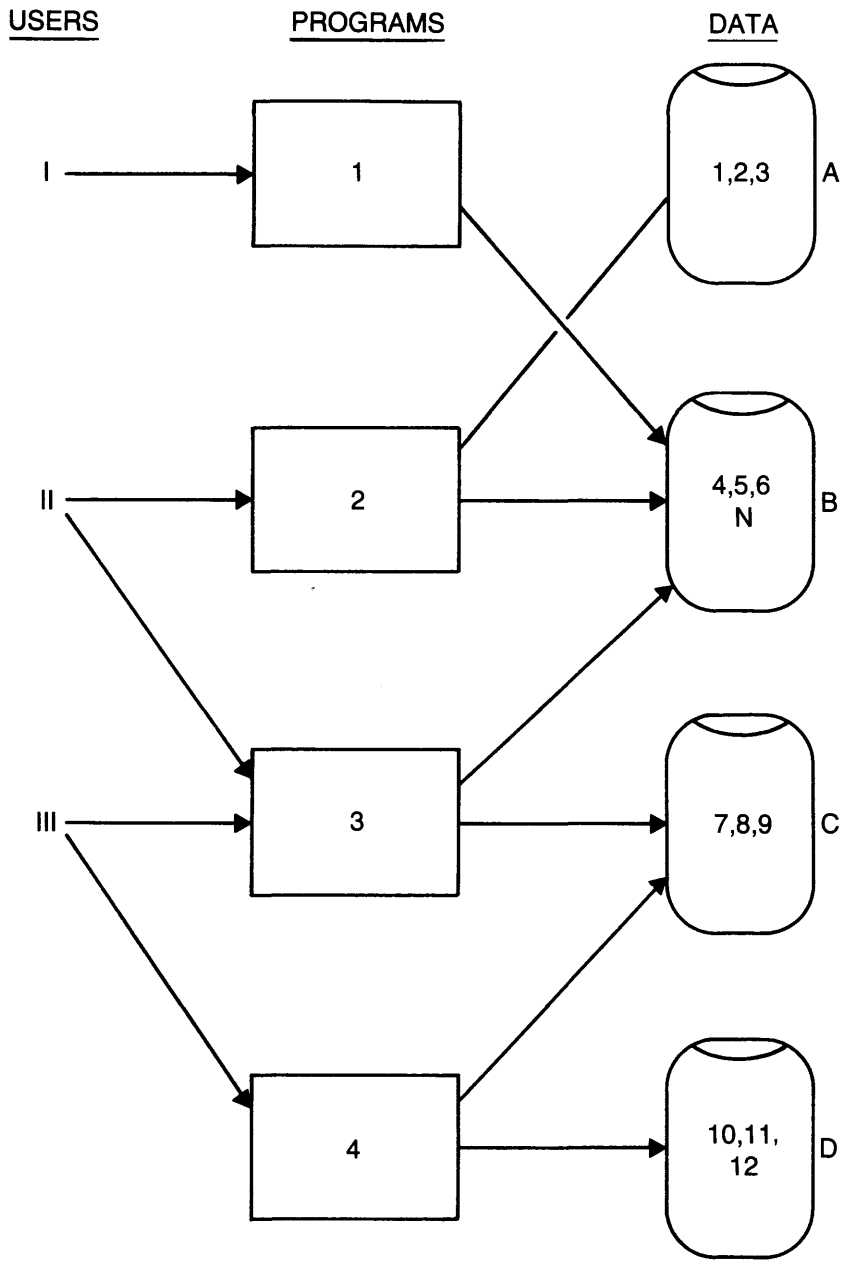
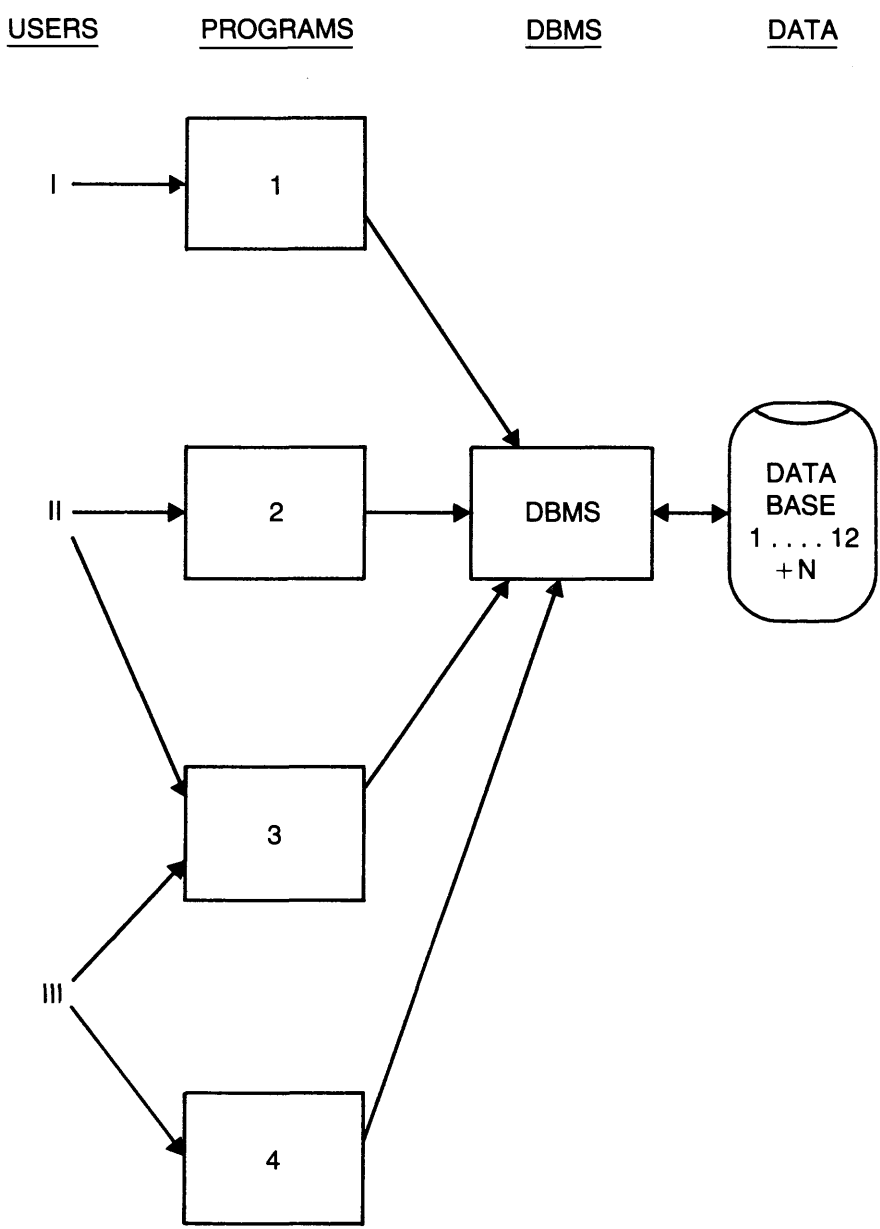


Figure 2
Data Base Environment Maintenance Cycle



and therefore must be activated consciously. If positive action is not undertaken, the features are not activated.

Resource Management

In the traditional data processing environment the ownership of data is normally based upon departmental requirements. Each user "owns" the data considered essential to that department, although the possibility exists that there is other data in the organization that each user would find useful. In the data base environment this additional data can easily be made available to other users. However, making this data available to many users is commonly referred to as data sharing and creates additional concerns about data integrity, security, and ownership that must be managed.

Consistency of Data Element Representation

The data base environment can facilitate a decrease in the number of times the same data are stored (data redundancy) and thereby increase the consistency of data. The data base environment provides an opportunity for ensuring that each data element is represented by the same specifications and attributes in all instances.

For example, a customer name will be represented consistently as thirty alphabetic characters in all applications. The same can be said about numeric data. As redundancy decreases, quality control of data entry and editing of transactions that will update the data base become important, since there may no longer be independently maintained files that can be reconciled to each other. On the other hand, editing is simplified because of the standardization of data elements.

Synchronization of the Updating of Data Elements

A by-product of the data base approach is that synchronization of update processing to master files may be easier to achieve. For example, with two independently maintained product files, it may be difficult to coordinate changes to stock-on-hand as goods are shipped and received. Order entry/processing is depleting stock, while production is replenishing it. With a single shared and coordinated product data base, the element stock-on-hand would exist only once and thereby eliminate the problem of ensuring synchronized update of two independent data elements on separate files. This contributes to improved accuracy and timeliness of information.

Components of the Data Base Environment

The primary component of the data base environment is the DBMS software, which is usually supplied by a commercial vendor. The installation of a DBMS and the ability of diverse users to access the data result in a greater need for coordinated management of the data resources that are contained within the data base. This management, often performed by a data base administration group, is another important component of the data base environment.

A software product that can assist in the management of the data base is called a data dictionary/directory system (DD/DS). A DD/DS maintains information about definitions, data elements, data records, relationships between records, and logical views of data for applications. This is often referred to as "meta data," and its availability facilitates a better environment for program maintenance, since there is less chance for misunderstanding what each data element represents.

Data Base Management Systems

DBMSs operate within the limitations and constraints of the hardware environment. A DBMS cannot be moved from the hardware of one vendor to that of another without transforming or rewriting the system to make it compatible, and the choice of a DBMS is limited by the operating system in place.

The DBMS typically provides data management services through three interrelated systems, which are often referred to as languages.

1. A data manipulation language (DML) is provided to afford application programs with a facility to interact with the data base, that is, add, modify, or delete either data or relationships. The DML is incorporated into the application programs and serves as the interface between the application program and the data that are stored by the DBMS.
2. A data definition language (DDL) is provided to define the structure and content of the data base. The schema (description of the entire data base) and sub-schema (logical view) are defined using the DDL. The schema specifies the name and other characteristics of data elements and their relationships. The sub-schema defines the specific logical views of the data required by the application program. Application programs can only access the data base via a sub-schema, and it is the sub-schema that limits the data ele-

ments and functions that an application can access and perform.

3. A storage structure definition language is provided to specify the physical location of the data elements on the storage devices where the data are maintained.

In most cases DBMSs use a number of additional software products that are an integral part of the DBMS. These include report generators that use special self-contained languages rather than conventional programming languages to create reports and query languages that can be used by non-EDP personnel for inquiry requirements. In addition, special "utility" routines are provided to reorganize the physical data base and remove deleted records.

There are also a number of facilities included in the DBMS that help maintain the integrity of the data base. Such facilities are usually provided in the form of auxiliary data services and include such items as recovery/restart routines, generalized edit and validate routines, and security and control features.

Data Base Administration/ Data Administration

Having multiple users of the same data increases the importance of centralized coordination of the use and definition of data and the maintenance of its integrity, security, and accuracy. Centralized coordination is usually performed by a group of individuals whose responsibility is typically referred to as data base administration; the individual who heads this function is typically referred to as the data base administrator. The data base administrator is generally responsible for the definition, organization, protection, and efficiency of the data bases, including defining the rules by which data is accessed and stored.

In some organizations the administrative and policy functions of data base administration have been separated from the technical functions. The policy and administrative functions are performed by the data administrator. The more technical functions remain the responsibility of the data base administrator. Some entities have expanded the role of the data administrator further to encompass the administrative and policy functions associated with data resource and information management. The data administrator function does not necessarily reside within the data processing group. Organizations sometimes put this important function in a separate corporate staff position.

Data base administration tasks also may be performed by organizational entities other than the data base administrator or data administrator. For example, a technical services group might perform the

technical data base design, an applications development group might perform the data base analysis and conceptual design activities, and the computer operations department might handle the day-to-day operation of the data base.

In a situation where the tasks of data base administration are not centralized, but are distributed among existing organizational units, the different tasks must still be coordinated if effective control is to be achieved. These tasks would typically include—

- Defining data elements and designing the data base.
- Maintaining data integrity, security, accuracy, and completeness.
- Coordinating computer operations.
- Monitoring and improving system performance.
- Providing administrative support.

Defining data elements and designing the data base. This includes how data is defined, stored, and accessed by users of the data base. In designing the data base, the processing, security, integrity, performance, and data requirements of the various users must be considered to ensure that the requirements are met on a timely basis.

Maintaining data integrity, security, accuracy, and completeness. This includes developing, implementing, and enforcing the rules for data integrity, completeness, and access. Issues include who may access data and how the access is accomplished; protection of the data base against inaccurate, invalid, or missing data; securing the data base from unauthorized access and destruction; and total recovery in the event of a loss.

Policies for disposition of the organization's data should be defined and enforced considering the regulatory requirements of the government and the organization's needs for historical data. Particular attention should be given to the retention of transactions that update the data base and the images of data base records either prior to or immediately after they are changed.

Consideration must also be given to ensure that complete and appropriate documentation is maintained dealing with the recording of procedures, standards, guidelines, and data base descriptions.

Coordinating computer operations. This includes assigning responsibility for physical computer resources and monitoring their use. Issues include ensuring that documented procedures are in place and are followed for operating the data base including scheduling of computer time, use of the system, and restart/recovery.

Monitoring and improving system performance. This includes developing performance measurements to continually monitor the level of service to users and the integrity of the data. Issues include monitoring performance, recognizing deficiencies, and taking steps to improve performance.

Providing administrative support. This includes coordination and liaison with the vendor of the DBMS, assessing new releases of the DBMS and their relative impact on the organization, and ensuring that appropriate internal education is provided.

Data Dictionary/Directory System

All DBMSs have an internal data directory to keep track of the various data elements. The DD/DS referred to in this discussion is an additional software tool designed to assist in managing the data base. This tool is optional. The DD/DS may be used to define the following:

- Data elements and their range of data values or specific data values
- Records
- Data elements associated with each record
- Relationships between records
- Logical views of the data elements for each application
- Physical location of data elements and records

Other information can also be included, such as output reports, input documents, application programs, authorized users, and other related information.

In almost all instances, the DD/DS uses a DBMS as the mechanism for storing, relating, and manipulating data and, consequently, uses the features of a DBMS. These features permit the user to request predefined reports about the DBMS, to formulate ad hoc queries, and to create special reports as needed.

Typical reports that a user might request include ones that—

- Identify the names and locations of data elements.
- Describe the data elements, (for example, alphabetic, numeric, field length).
- Identify the organization or person responsible for the integrity and accuracy of each data element.
- Define the edit and validation rules.
- Identify programs and transactions associated with a data element.

Figure 3
Active Data Dictionary/Directory System

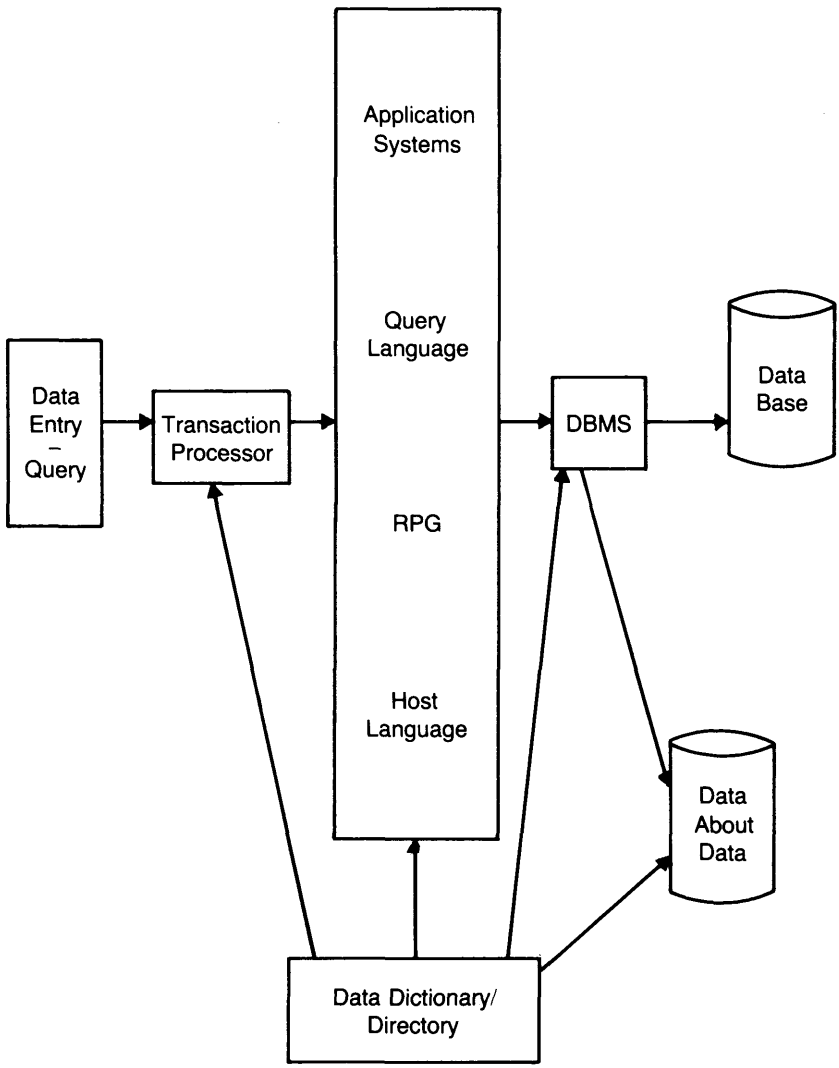
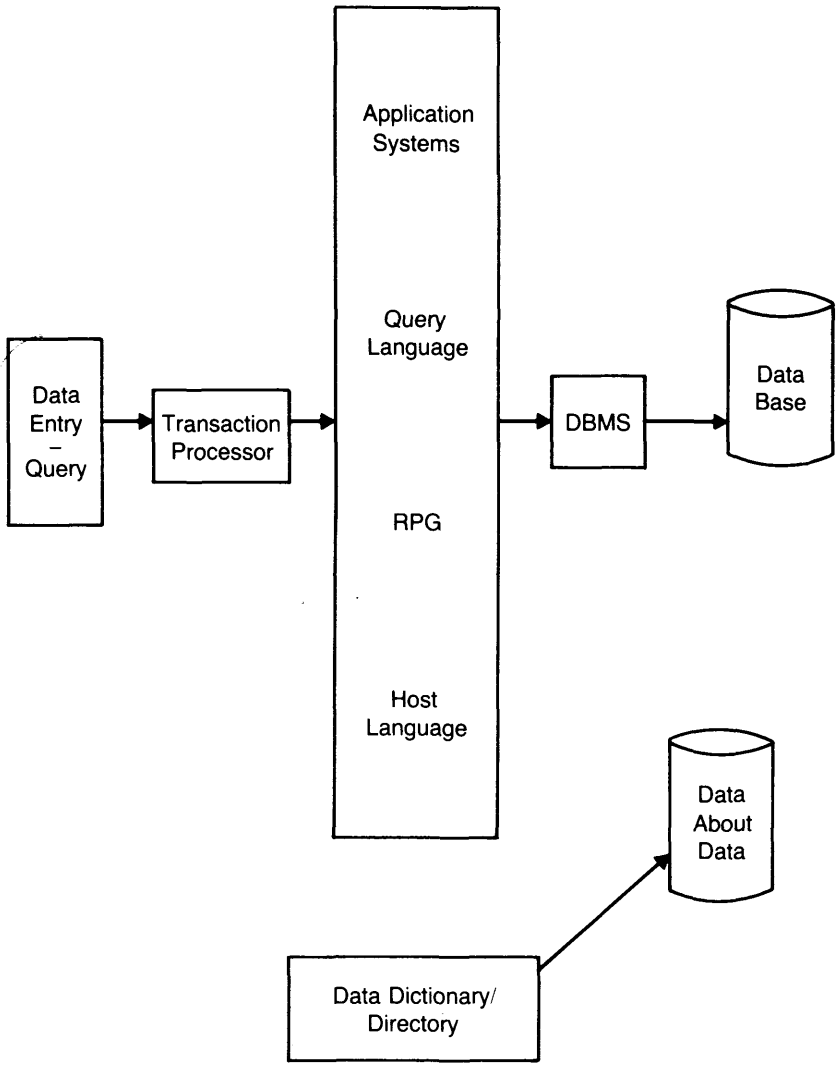


Figure 4
Passive Data Dictionary/Directory System



The DD/DS assists the data base administration function in performing the activities of data base development (data collection and analysis and reduction) and can serve as the mechanism for establishing standardized documentation and definitions of the data base environment.

In addition to storing and reporting information regarding the data base environment, some DD/DSs also create the input for updating the internal data directory system that the DBMS uses to support the various views of the data elements. This is called an active DD/DS. This active feature enhances data integrity and security by ensuring that the information stored in the DD/DS is identical to that used by the DBMS's internal data directory system.

Other DD/DSs, termed passive systems, are updated separately from the creation of the directory system that the DBMS uses to support the various views of the data elements. From a control point of view, having a DD/DS is desirable, and the active feature enhances reliability.

Figures 3 and 4 illustrate these two types of DD/DSs.

Summary

A data base environment is characterized by two principal features: the sharing of data by multiple users within the organization and the use of a DBMS. The important elements of this environment include—

- A DBMS for establishing data relationships, access paths, and features to control security, integrity, and access to data.
- Data base administration to coordinate management of data resources.
- A DD/DS for keeping track of data definitions and data elements.

This environment can affect the system of internal accounting control and may require the auditor to modify planned audit procedures. Chapter 2 describes those control considerations in a data base environment that the auditor should be aware of when planning the audit.

2 Control Considerations in a Data Base Environment

To provide effective control over data processing, management should identify the relevant control concerns or exposures and implement appropriate control procedures that are cost-beneficial. In a data base environment the basic control considerations do not change. However, because of the differences in a data base environment attributable to data independence and the sharing of data among diverse users, the control procedures may differ from those that might be employed in a non-data base environment. Understanding the implications of these different control procedures should aid general management and auditors in evaluating control requirements in a data base environment.

There are three broad areas where control in a data base environment may differ from that encountered in a non-data base environment: access/update, coordination of activities, and concentration of resources.

Access/Update

Data in a data base environment are accessed through data requests by the application programs to the DBMS or through the special report-writing facilities of the DBMS and DD/DS, including interactive query languages. Access/update control considerations include retrieving data elements, updating data elements, and executing functions concurrently. The necessity for additional or different controls in the data base environment is influenced by the sharing of data by users and by the greater availability of the data elements to the user as a result of on-line access and easy-to-use query languages.

Retrieving data elements. Controls should ensure that users can access only those data elements that are authorized for that user.

Retrieving data elements includes the capability to initiate access to data elements based on direct inquiries (such as employee X's pay rate), as well as by indirect inquiries (such as the employee identifi-

cation number for each employee whose pay rate exceeds ten dollars per hour and who has been employed less than one year).

Updating data elements. Controls should ensure that each user can update only those data elements that are authorized for that user.

Updating data elements includes the capability to change the contents of a data element, add or delete a data element, or change the relationships between data elements (for example, data element A, which was linked to data element B, now will be linked to data element C).

Executing functions concurrently. Controls should ensure that if there are two or more authorized programs (transactions) requesting to update data at the same time, the result should be accurately reflected without loss of data. Further, if one authorized program requests an update and one or more other programs request retrievals at the same time, the resulting information to end users should be accurate and consistent.

Due to the sharing of data between user programs that could be executing at the same time, two programs might request update/retrieval functions of the same data at the same time. If not properly controlled, this could have an adverse effect on the integrity of the data. Therefore, the DBMS procedure for accessing data should essentially "lock" the group of shared data until the update processing is complete before allowing the retrieval function to access the data.

Commercial DBMS software provides built-in safeguards to prevent the loss of data at a time when there is more than one request to access a particular data element. This requires defining and establishing a logical order for handling queuing requests.

Coordination of Activities

The sharing of data by diverse users requires coordination of activities to achieve an appropriate level of controls, a consistent interpretation of the meaning of the data elements, and the proper timing of update and cutoff of transactions.

Appropriate level of controls. The level of control over each data element should be commensurate with its most sensitive and critical use.

Shared use of data resources in a data base environment does not imply equality of requirements among the users. For example, data on the quantity of each style of a particular inventory item on hand may be of critical importance to a marketing user but of virtually no impor-

tance to an accounting user, who may only be concerned with the total quantity. Users with different logical views are apt to require varying levels of control over the data elements. The requirements of the most critical user should be of primary concern in the design of the control system.

If incorrect data are accepted into the data base and either subsequent input validation or internally generated transactions are dependent on that data, the overall integrity of the data base can be affected in ways that are difficult to identify. Erroneous data entered into a traditional data processing environment could create similar conditions, but with the technical complexities and the integration of data in data base systems, the extent of error may be more difficult to isolate and correct because of the number of users that may have used the erroneous element.

Consistent interpretation of data definition. Controls should ensure that the descriptions of data elements are coordinated to ensure consistent interpretation of meaning and appropriate use of the data elements.

With many diverse users of data elements, procedures should be implemented to ensure a consistent and accurate understanding of the correct definition of each data element and the purpose for which it is to be used. For example, does a price of a product mean wholesale or retail? Consistency of interpretation requires coordination and communication among users.

Proper timing of update and cutoffs. Controls must ensure that cutoffs are coordinated so that all user requirements for information during or at the end of a particular accounting or operating period are satisfied.

The sharing of data among applications and across organizational groups might create conflicts with regard to processing cycles and selecting cutoff times. For example, pricing data often change in dynamic market situations, yet information appearing on cost accounting, marketing/sales, and billing reports must reflect accurate pricing consistent with the time periods in question. Coordination among users is required to ensure timely cutoff of information and the availability of historical positions as necessary.

Concentration of Resources

There are control considerations that are associated with the concentration of resources in the following areas.

Physical concentration. Controls should ensure that the physical data structure is protected from improper alteration and destruction.

Common sharing of data using a single physical representation means that multiple users now rely upon the continued availability of that single representation. Physical loss of data through fire or data storage device failure can prevent use of the physical representation and can affect many users.

Data structure representation. Controls should ensure that the multiple logical views are maintained and are accessible only to the authorized users.

Sharing data through multiple logical views from a single physical representation implies using complex data relationships. This, in turn, means that shared data bases could have extensive indices and pointer structures that must be available to support the access to data.

Reliance on DBMS. Controls must ensure that the DBMS is operated in an appropriate manner to maintain data integrity.

In a data base environment there is reliance on the proper and continued functioning of the DBMS software. Failure of the DBMS software will have a significant effect on the availability of data. The greater the complexity of any software, the greater the possibility of failure.

Reliance on key personnel. Controls must ensure that the organization has a sufficient number of personnel trained to use the DBMS and related application systems.

An adequate number of trained personnel reduces the dependency of the organization upon a few individuals whose departure could disrupt the operations of the DBMS and related applications.

In a data base environment traditional application-specific controls are often transferred to a more macro level under control of the DBMS. The responsibility to ensure that these controls are appropriate as well as the responsibility for data base design and security is usually assigned to the data base administration function. This places more authority and responsibility in the hands of fewer people. The risks of turnover, incompetence, and compromising behavior take on greater significance.

Summary

A data base environment will introduce the need for additional control over access/update, coordination of activities, and concentration of resources. It is important that management and the auditor recognize the importance of these new considerations, the need for additional controls, and the impact that an absence of control may have upon the integrity of the data.

3 Audit Considerations in a Data Base Environment

This chapter suggests methods for gaining an understanding of application systems in a data base environment and identifies some new or revised control techniques and audit procedures. Audit procedures that are normally used in connection with EDP systems are not dealt with in this chapter unless the data base environment requires a change in emphasis or application.

The data base environment affects audit procedures primarily in gaining an understanding of the system, identifying new or revised control techniques, and designing and conducting compliance and substantive tests.

Gaining an Understanding of the System

General Guidelines

The introduction of data base technology does not necessarily have an effect on conducting an audit. However, upon determining that a DBMS has been implemented, a preliminary assessment of its impact on the audit can be determined by considering the following factors:

- The greater the number of transaction types that update the same financially significant data elements, the greater the impact of the data base environment on the financial environment.
- The greater the number of diverse users that share financial accounting data, the greater the impact of the data base environment on the financial environment.
- Transaction types such as deletes and updates have a greater impact than reads on the financial environment.
- DD/DSs that are accurate representations of the content or structure of the data base can substantially reduce the effort required to understand and evaluate the data base environment.

- The greater the number of diverse users that share the data (the level of data sharing), the greater the need for coordination of activities and an effective data base administrative function.
- The placement of the data base administration function within the company can provide an indication of the level of data sharing. If data base administration is a part of application programming, then the DBMS is probably application-specific, and the DBMS is being used as an access method with little or no data sharing. If data base administration is a part of technical services, then the DBMS probably supports multiple applications, but the level of data sharing is probably low. However, if data base administration reports to a high-level officer, usually there is a high level of data sharing.

Migration of Control

Processing controls that were typically included in the application programs may have migrated to the DBMS. The auditor should determine the controls that have migrated to the DBMS and the degree of data sharing involving data elements of financial significance.

There are control features available in the DBMS that can be used to satisfy certain control functions. These features include the following:

- Edit and validation facilities
- Limitation of access to the data base
- Checkpoint/restart procedures
- Control file balancing

If these features are implemented through the DBMS, the associated controls that normally would have been found in the application program or user department may now be part of the DBMS; that is, the control features will have migrated from the application system to the DBMS. For example, the DBMS can provide comprehensive instructions to perform edit and validation tests of each transaction as it is initially entered into the system. In a traditional data processing environment the edit and validation tests are normally applied within an application program and therefore may be different for each system through which the same transaction is processed. The transfer of control functions to the DBMS has the advantage of requiring only one set of controls to service all application programs accessing the same data and ensures that the data from all applications are subject to the same controls.

If control has migrated to the DBMS, the auditor cannot gain a complete understanding of control at the application level but must

spend more time with data processing personnel responsible for implementing and maintaining controls at the data base level.

Level of Data Sharing

To determine the effect of data sharing on the audit the auditor should establish—

- The data elements of financial accounting significance that will ultimately affect the audit opinion.
- The transactions that create, amend, or delete those elements.
- The departments or persons who have the ability to execute those transactions.

Having accomplished this, the auditor can determine the exposure to error that may exist and the nature of control necessary to mitigate those exposures.

The auditor can determine the level of data sharing by using a transaction impact matrix, using a DD/DS, and examining schemas/sub-schemas.

Transaction impact matrix. A transaction impact matrix (table 1) is a graphic technique that can be useful when analyzing a small number of transactions and data elements. A matrix is constructed with one axis representing elements and the other axis representing transactions. In the following example of a bank demand deposit accounting system, the cells contain either an "R," which means read only, the data can only be accessed and not changed, or a "U," which means update, the contents in the data element could be changed, added, or deleted or the relationships could be amended.

Table 1
Transaction Impact Matrix
(Customer Accounts)

	Elements			
Transactions	*Account Balance*	*Customer Number*	*Account Number*	*Date of Transaction*
Withdrawal	U	R	R	U
Deposit	U	R	R	U
Inquiry	R	R	R	R
Add a new account	U	U	U	U

R = Read
U = Update

The value of the transaction matrix can be further enhanced by adding timing (frequency) and volume information. This information provides insight into the level of concurrent data sharing, that is, multiple transactions updating the same data element at the same time.

Data dictionary/directory system. DD/DSs generally have "where used" reports. These are reports that indicate which applications or transactions affect a data element. Thus, once a data element of accounting significance has been identified, these reports can be used to identify all applications or transactions that can affect the particular data element. In addition, for any specified application or transaction, the associated data elements and how they are affected by that transaction can be determined using these reports. In some DD/DSs, a scanning routine can be invoked to list all application programs in which a data element appears and the exact transactions that may affect the data element.

Evaluation of schemas/sub-schemas. If there is no DD/DS available and the system is complex, the impact of transactions on data elements can be analyzed by evaluating the schemas/sub-schemas of the data base. This method would require the auditor to obtain schema/sub-schema listings from the DBMS and analyze the entries in terms of their meaning and processing logic. From this information the auditor would construct the logical equivalent of the DD/DS "where used" reports to determine which sub-schemas and transactions have an impact on the data elements of accounting significance and the functions being performed by each transaction.

Sources of Information

The data base administrator or the organizational entity responsible for data base administration is often the only reliable source of information about the data base environment. The data base administrator, or equivalent, usually is responsible for defining, designing, implementing, and maintaining the data base. Usually the data base administrator also is responsible for maintaining adequate documentation of the data base environment. This information and the data base administrator's day-to-day experience can prove to be a valuable resource.

The DD/DS is another source of information for the auditor. It contains information about the data elements and record types, such as the following:

- Name
- Size

- Edit criteria
- Owner
- Source
- Frequency of use of data elements and record types
- Relationships between components of the data base

Many DD/DSs have report-writing or query features that allow the auditor to request reports showing details of data element relationships and other pertinent information.

If a separate DD/DS is not available, the basic information that is required can still be obtained. DBMSs have associated with them data definition language processors. The input to the data definition language processor is a description of all the data elements, record types, and relationships. Part of that description is a definition of the attributes, such as size and name. This information can also be used to perform transaction impact analysis.

Identifying Control Techniques and Designing Audit Tests

In a data base environment the auditor will typically encounter certain new or revised control techniques that will affect the overall evaluation of internal control and the degree of reliance that can be placed on controls.

This section describes certain new and revised control techniques. It provides a description of each technique, the control functions affected by the technique, and a listing of possible audit procedures to test the control.

In table 2 (on page 26) control techniques are grouped by access/update controls, system design controls, data base administration controls, and operational control. The control techniques are then related to the control functions that they are designed to address.

Access/Update Controls

Restrict Data Resources and Update Functions to Authorized Users by Passwords

Passwords can be used to restrict user access to the data base. These restrictions could apply to individuals, organizational units, terminals, and programs or transactions. Restrictions also can be implemented to restrict the use of functions, such as read, update, modify, or delete. Data resources at the logical level can be restricted in terms of record types (payroll record) and data element types (pay rate) as well as

Table 2

Control Techniques and Related Control Functions

Techniques	*Functions*		
	Access/ Update	*Coordination of Activities*	*Concentration of Resources*
Access/Update Controls			
Restrict data resources and update functions to authorized users by passwords	X		
Restrict data resources and update functions to authorized users by sub-schema	X		
System Design Controls			
Implement application systems using a systems development life cycle tailored to consider the use of a DBMS		X	X
Implement a standardized approach for making modifications to application systems		X	X
Automatically generate data base description by the DD/DS		X	
Develop adequate transaction trails		X	
Consider the DBMS return codes during the detailed design phase	X	X	X
Data Base Administration Controls			
Assign responsibility for data ownership	X	X	
Centralize administration of schema/sub-schema	X	X	
Maintain adequate segregation of duties	X		X
Operational Control			
Analyze internal storage structures (pointers)			X

specific record values (John Doe's payroll record) and data element values (John Doe's pay rate).

A password identifies a user, and the DBMS uses this identification to restrict any or all of the following:

- Use of terminal
- Programs that may be executed
- Functions to be performed (read only, update, and so on)
- Data elements available

For passwords to be effective, adequate procedures must be enforced for changing passwords, maintaining secrecy of passwords, and reviewing and investigating attempted security violations.

The control function addressed is access/update. Using passwords and logically relating them to terminals, programs, and subschemas help to ensure that only authorized users can enter, amend, or delete data. Transactions that were intended to be restricted may not be restricted if—

- Passwords are not implemented.
- Passwords are implemented but a proper relationship has not been established between the password and terminals, transactions, programs, and data element use and access.

Possible audit procedures

- Review and test the security profile. The security profile is similar to the transaction impact analysis and is a matrix showing which user (or user group) can read or update which data elements. The cells contain information about the functions that the user may perform with respect to the data element. For example, in a bank system tellers may be authorized only to view account balances and to enter data from routine transactions (deposits, withdrawals, transfers). They may not be authorized to modify or change data that has already been entered. The security profile provides a summarized overview of who is authorized to change and modify what data. The security profile can then be compared to the transaction impact analysis to determine that only authorized users can change or modify the data.
- Review and test compliance with standards and procedures for establishing and maintaining the security profile considering the roles of the EDP department and users and how they coordinate development of appropriate security profiles.
- Examine the data definitions language specifications including the description of the data base and its logical views to determine if

only authorized transactions are being permitted to access the data.

- Process sets of actual transactions to determine if the security mechanism is operative. If an attempt is made to enter an unauthorized transaction into the system and it is accepted by the system, care must be taken to reverse the entry.
- Examine access logs to determine if unauthorized users are attempting to access the data base and that procedures exist to review the logs on a regular basis and to investigate attempted unauthorized accesses.

Restrict Data Resources and Update Functions to Authorized Users by Sub-Schema

The sub-schema that describes an application program's logical view can be used as a mechanism for limiting the program's access to and operations on the data. This control technique is accomplished by requiring an application program to access the data base using a sub-schema that is specifically designed to give the application program only those data elements it requires to accomplish its processing task. In addition, what the program can do with each data element can be specified through the sub-schema, such as read only, update, delete, and so on. This restriction of application programs to predefined data access and functions is an important control feature found in many DBMSs. This technique is less effective for certain types of DBMSs in which the sub-schema is imbedded in the data manipulation language and is therefore under program control.

The control function addressed is access/update. Using sub-schemas to restrict access helps to prevent unauthorized transactions from gaining access to the data base and thus helps ensure the accuracy and maintenance of the data.

Possible audit procedures

- Review and test the security profile.
- Review and test compliance with the standards and procedures for establishing and maintaining the security profile and in particular that programs are restricted to data elements and functions required by the programs.
- Examine the data definition language specifications including the description of the data base and its logical views to determine if only authorized transactions are being permitted to access specified sub-schemas of the data base.
- Process actual transactions to determine if the security mechanism is operative.

Although available DBMS protection features may appear effective, they have a fundamental weakness: DBMS access controls may be violated by using available utility programs that permit direct access to the data base. Controls over use of utilities need to be considered in the evaluation of a data base environment just as they should be considered in a traditional EDP environment. This exposure can be partially offset by restricting the programs that can be run on the computer when the DBMS is loaded or by monitoring the instances in which these DBMS utility programs are being used.

System Design Controls

Implement Application Systems Using a Systems Development Life Cycle Tailored to Consider the Use of a DBMS

Systems development life cycle (SDLC) is a technique for introducing discipline and structure into systems development and maintenance. It represents a formalized, step-by-step approach that requires adherence by all application project teams. While the use of an SDLC is appropriate in both data base and in non–data base environments, its usefulness in a data base environment can be enhanced by tailoring it to provide for the incorporation of a data base design methodology, which is a detailed description of the activities that are performed to create data bases. There are three major phases of data base design: conceptual, logical, and physical.

Conceptual design includes the structure and relationships that exist between entities, (for example, branches, orders, customers) and is not concerned with the structural requirements and design constraints of individual DBMSs.

Logical design includes mapping the conceptual design to limitations of individual DBMSs as well as making trade-offs required to optimize performance, integrity, or security features.

Physical design includes mapping the logical design to the hardware. This entails making hardware-related performance trade-offs such as channel balance, disk utilization, and input/output configuration.

These three design phases are related to requirements definition, general system design, and detail design in the traditional SDLC.

The SDLC should also take into consideration the structure of the data base, the record types and their interrelation, all of which can affect the procedural logic of the program, and access paths required.

Finally, the SDLC should be able to use consistent edit and validation criteria associated with each data element. There are two methods of ensuring consistency: (1) duplicate the same edit and validation code for each program updating a target data element or (2) design

the edit and validation procedures into a common routine through which all transactions must pass and incorporate this routine into the system.

Control functions addressed are coordination of activities and concentration of resources. Using an SDLC designed specifically for the data base environment helps to ensure accuracy and integrity because the processing of each transaction that affects a data element is designed according to specified procedures. These procedures take advantage of the particular DBMS's features related to consistency of edit and validation, and restriction of resources and functions through sub-schema definition.

Possible audit procedures

- Review and test standards and procedures for system development and data base design to determine if they are appropriate for a data base environment. Particular attention should be given to—

  Using common edit and validation routines.

  Structuring record types and their interrelationships.

  Centralizing coordination of responsibility for documentation and procedures.

  Using sub-schema features to restrict access and functions of programs.

- Examine system development documentation and determine if the project has followed standards. Specific documentation to be examined would include data base design, sub-schema authorization, and review for security.
- If there is a DD/DS, review and substantiate the accuracy of its contents with respect to edit and validation rules, security profiles, and maintenance procedures.

Implement a Standardized Approach for Making Modifications to Application Systems

Control is enhanced by standardizing and documenting each modification. This would include performing an impact analysis of new or existing transactions on the data base each time a modification is required. The resulting analysis would indicate the effects of the changes on the security and integrity of the data base.

Control functions addressed are coordination of activities and concentration of resources. Implementing a standardized approach to

modify application systems is a technique that can help prevent degradation of the accuracy, integrity, and completeness of the data base. Without a standardized approach, the data base structure and content could be changed or altered without proper management authorization.

Possible audit procedures

- Review and test compliance with the standards for data base modification, including the role of the data base administration function, to assess whether the standards result in a changed data base that will properly support new and existing transactions and applications.
- Review impact analysis to determine whether inappropriate relationships exist for data elements of accounting significance.
- Review and test DD/DS update and change procedures to identify and evaluate conditions that could adversely affect the accuracy and completeness of the data.
- Compare the formal documentation with the actual data base descriptions to determine the accuracy and completeness of the documentation and therefore the degree of reliance that can be placed upon the documentation as a source of reliable information.

Automatically Generate Data Base Description by the DD/DS

The DD/DS can be used to automatically generate data descriptions and logical views of the DBMS. The data base designer describes the structure of the data base in the DD/DS. This information is then extracted, reformatted, and used by the DBMS. Alternatively, the DD/DS and DBMS can share the same description of the data base, and the DBMS substitutes the description of each data element in the DD/DS object code for its own embedded directory.

The control function addressed is coordination of activities. This technique helps ensure that the DD/DS and data base include identical information and therefore allows the auditor to use the more understandable output from the DD/DS in assessing the transactions that affect data of accounting significance.

Possible audit procedures

- Review and test compliance with the standards for describing the data base.
- Examine DD/DS contents, data base descriptions, and logical views and compare for equivalency. This is essential when the DD/DS is passive (see chapter 1).

Develop Adequate Transaction Trails

The transaction trail is management's and the auditor's view of what has happened to the data base. It is an historical record of the accesses and changes that have occurred to the value of a data element, including information about when the change was made and by whom. In a data base environment the value of a data element may be changed by many types of transactions and a correspondingly large number of applications and users. In the data base environment the creation of the transaction trail has often migrated to the DBMS. As part of standard restart/recovery features, the DBMS typically creates a journal log of all data base access transactions (updates, reads, and the like). The journal log also identifies the source and time of access. In addition to the journal log, a DBMS can typically also create before and after image logs. An image log is a record of the physical data elements just prior to or just after an update that is stored by the DBMS for a predetermined period of time.

The journal and the before and after image logs represent transaction trails that can be used by the auditor. The auditor could identify all the transactions that affect the data elements of interest to the auditor; these transaction types could then be extracted from the logs for further examination and comparison to the authorized transaction types.

The transaction audit trail for a data element is the sum of the transaction trails of all transactions that affect that data element. A transaction may be involved in many data element transaction trails.

The control function addressed is coordination of activities. Developing adequate transaction trails helps ensure that the integrity and accuracy of the data base is maintained. If the transaction trails are not periodically monitored and analyzed, potential degradation of integrity and accuracy could go undetected.

Possible audit procedures

- Review data base administration standards and procedures for transaction trail creation, the period of time transaction trails are retained, and restart/recovery transaction requirements and evaluate their adequacy.
- Reconcile the transaction trails to actual data base contents. This may be accomplished by utilizing the restart/recovery facilities of the DBMS.

Consider the DBMS Return Codes During the Detailed Design Phase

Return codes are a means by which the DBMS communicates with the application program. An application program can use return codes to determine if transactions have been executed correctly. If the trans-

action/DBMS interface is not controlled and the application programs do not use return codes correctly, the integrity of the data base may be compromised. For example, if a transaction did not terminate normally, and this was not dealt with properly by the application program, the integrity of the data base could be threatened.

Control functions addressed are access/update, coordination of activities, and concentration of resources. This technique helps ensure the complete and accurate update of the data base.

Possible audit procedures

- Review programming standards and procedures to determine if standards exist to ensure that return codes are being adequately evaluated by the application programs.
- Examine application program codes to determine if standards have been followed.

Data Base Administration Controls

Assign Responsibility for Data Ownership

In a data base environment many individuals may share the same data through their own logical views and may perform various functions on the data elements. In circumstances where many individuals can affect the accuracy and completeness of a data element, a clear and definite assignment of responsibility for the accuracy and integrity of each data element is required. A single user should be assigned responsibility for defining the logical meaning for each data element and the access and security rules, such as who can use the data element (access) and what functions they can perform (security).

Control functions addressed are access/update and coordination of activities. Assigning specific responsibility for ownership helps to ensure the integrity of the data base. If several users are able to make decisions affecting the accuracy and integrity of data elements, the likelihood of the data elements becoming corrupted or improperly used increases. This problem normally increases with the degree of data sharing.

Possible audit procedures

- Review the standards, procedures, and guidelines, including the role of the data base administrator, to determine if "owner" authorization is required prior to granting to others a sub-schema involving use of the owner's data.
- Sample some logical views containing data elements of accounting significance and ensure there is an owner identified and that the

owner has approved any other use or access to the data elements. Documentation of the owner's review and approval should exist.

- Observe modifications made to the logical views of the data base and ensure that all authorization procedures have been carried out and documented before amendment of the sub-schema.

Centralize Administration of Schema/Sub-Schema

The creation and modification of sub-schemas (logical views) should be performed by a single organizational unit. Application programmers should be restricted in their access to the schema, sub-schema, and data base. Centralized control over the administration and amendment of schemas/sub-schemas helps to prevent unauthorized "fixes" or "patches" to schemas and sub-schemas, the creation of unauthorized sub-schemas that are not documented in the DD/DS, and the unauthorized sharing of sub-schemas (that is, two applications using the same sub-schema when only one was authorized and the other has "borrowed" the sub-schema).

Control functions addressed are access/update and coordination of activities. Control over creation and modification of the schema/sub-schema helps ensure the accuracy, integrity, and correctness of the data. If this control were not implemented, knowledge of the data base structure could proliferate and lead to unwanted usage of the data base and possible increased access to the data base. This situation can occur when there is no control over programs' access to sub-schemas, poorly designed sub-schemas are used (for example, the application program receives more data than necessary to perform its tasks), or where specific control over access is not implemented (for example, programs are not required to have unique logical views).

Possible audit procedures

- Review standards, procedures, and guidelines for the creation and implementation of schemas/sub-schemas and subsequent modifications to ensure that application programs require a unique sub-schema or to determine that authorization has been obtained from users who share data where sharing is considered efficient.
- Examine documentation of sub-schema generation to determine that proper authorizations were received and that the data transferred to the programs were only what was required to carry out their processing tasks.

Maintain Adequate Segregation of Duties

The responsibilities for performing the various activities required to design, implement, and operate a data base environment are divided

among technical, design, and administrative personnel including the users. Their duties include system design, data base design, administration, and operations.

Control functions addressed are access/update and concentration of resources. Maintaining adequate segregation of duties is necessary to ensure the integrity, correctness, and accuracy of the data base.

Possible audit procedures

- Examine job descriptions, standards, and procedures for those responsible for technical support, design, administration, and operation of the data bases to ensure their responsibilities are not incompatible.
- Observe individuals performing data base design, development, and modification and review written evidence indicating who performed the tasks to ensure that stated procedures are followed.

Operational Control

Analyze Internal Storage Structures (Pointers)

The DBMS keeps track of data by using internal storage structures (pointers). Analysis would involve determining that pointers do, in fact, represent actual addresses and valid current data. In large data bases the number of pointers and the size of the data structures make the checking of internal data structures difficult and could cause a strain on system resources if done at one time. However, checking that pointers are valid could be done on a cycle basis so that the entire data base is reviewed over a period of time. This analysis is especially important in those data bases in which cross-reference information is only represented one time rather than in each record where it might be important. For example, the customer number can be included in each order or the system can rely completely on pointers from the record to each order.

The control function addressed is concentration of resources. This technique helps to prevent loss of structural integrity and data.

Possible audit procedures

- Review and assess the procedures and methods employed by the EDP department to ensure pointer integrity and the frequency of review.
- Review documentation of procedures to ensure pointer integrity analysis has been performed on a periodic basis by the DBA or other appropriate individual.

- Examine storage structure analysis reports to determine if appropriate action has been taken when there has been a loss of integrity of the internal storage structure (for example, pointers with invalid addresses). Such reports are normally generated by the DBMS, usually through the special utility programs provided with the DBMS.

Accessing DBMS-Managed Data Bases

The auditor may need to access DBMS-managed data bases to conduct audit tests. Accessing such data bases could require modification to existing techniques and existing audit software. The modifications result from the change in data structure. There are hierarchies or networks to be "navigated" to access data in the form and content required for audit. For example, a data base environment may consist of the data structure illustrated in figure 5.

If the auditor requires a listing of customers and the transactions associated with each, retrieving this information directly would not be possible. To retrieve the information the auditor would have to navigate or traverse the data base; that is, for each customer the related accounts would have to be retrieved and for each account the associated transactions would have to be retrieved. There may be no way to go directly from customer to transactions.

Balancing customer accounts would also require navigating and retrieving data. If an auditor were to attempt to reconcile customer account balances with detailed transactions associated with the balance, in a data base the auditor would require—

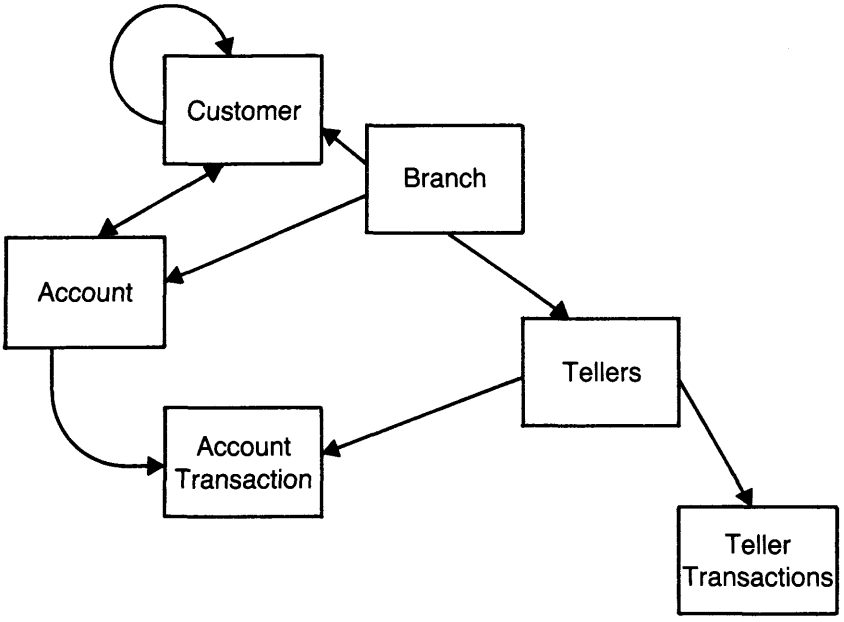
- The content of the record types found in the particular sub-schema about which the auditor was concerned. This would include data elements and their descriptions.
- The structure of the data base that is needed to develop a retrieval strategy.
- Transactions that affect data elements that are required to determine if—

  The data elements originate from the same source or have been independently created.

  The timing of updates is synchronized.

Once this basic information has been obtained, the auditor would still have to design, test, and perform the audit software routines to obtain access or extract the data for the audit procedure. The existence of a data base environment does not affect the auditor's objectives or

Figure 5
Data Structure



audit procedures that would be performed once the data is obtained, but it does complicate the means of retrieving the data and the factors that could influence the correctness and completeness of the information. The auditor should also be cognizant of the reliance that is being placed on the DBMS to extract the information and should have a basis for relying on the integrity of the DBMS software.

There are a number of alternative strategies that may be used by auditors in gaining access to data managed by a DBMS, including—

- Converting the data base into an extracted sequential ("flat") file. This involves reading the data base using a client program or utility to read and reformat the data into a sequential organization suitable for access by the auditor's audit software package. Often, files of this nature are created as part of the client's normal back-up procedures and can be made available to the auditor with minimal effort.
- Using special purpose interfaces between conventional audit software packages and the DBMS.
- Using the available query and report writer languages available with most DBMS products to interrogate or extract files.

Summary

The data base environment adds complexity to conducting audits. There are several different data structures and access concepts, control techniques, and system software components that may now affect processing of significant accounting transactions. If it is decided that a review of the data base environment is required, then the auditor should recognize that the nature of the review is likely to be more complex; the review time is likely to increase; and additional skills and knowledge are likely to be required. In most instances, the auditor will need to obtain specialized technical training or assistance from a trained EDP auditor to perform an adequate review of a significant application system operating in a data base environment.